ORACLE'S OPPOSITION TO GOOGLE'S RULE 50(A) MOTION & RESP. TO ECF NOS. 1934 & 1955

REDACTED VERSION OF DOCUMENT SOUGHT TO BE SEALED

ORRICK, HERRINGTON & SUTCLIFFE LLP
KAREN G. JOHNSON-MCKEWAN (SBN 121570)
kjohnson-mckewan@orrick.com
ANNETTE L. HURST (SBN 148738)
ahurst@orrick.com
GABRIEL M. RAMSEY (SBN 209218)
gramsey@orrick.com
405 Howard Street, San Francisco, CA 94105
Tel: 1.415.773.5700 / Fax: 1.415.773.5759
PETER A. BICKS (*pro hac vice*)
pbicks@orrick.com
LISA T. SIMPSON (*pro hac vice*)
lsimpson@orrick.com
51 West 52nd Street, New York, NY 10019
Tel: 1.212.506.5000 / Fax: 1.212.506.5151

BOIES, SCHILLER & FLEXNER LLP
DAVID BOIES (*pro hac vice*)
dboies@bsfllp.com
333 Main Street, Armonk, NY 10504
Tel: 1.914.749.8200 / Fax: 1.914.749.8300
STEVEN C. HOLTZMAN (SBN 144177)
sholtzman@bsfllp.com
1999 Harrison St., Ste. 900, Oakland, CA 94612
Tel: 1.510.874.1000 / Fax: 1.510.874.1460

ORACLE CORPORATION
DORIAN DALEY (SBN 129049)
dorian.daley@oracle.com
DEBORAH K. MILLER (SBN 95527)
deborah.miller@oracle.com
MATTHEW M. SARBORARIA (SBN 211600)
matthew.sarboraria@oracle.com
RUCHIKA AGRAWAL (SBN 246058)
ruchika.agrawal@oracle.com
500 Oracle Parkway,
Redwood City, CA 94065
Tel: 650.506.5200 / Fax: 650.506.7117

Attorneys for Plaintiff
ORACLE AMERICA, INC.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

ORACLE AMERICA, INC. Plaintiff, v. GOOGLE INC. Defendant.	Case No. CV 10-03561 WHA **ORACLE'S OPPOSITION TO GOOGLE'S RULE 50(A) MOTION & RESP. TO ECF NOS. 1934 & 1955** Dept.: Courtroom 8, 19th Floor Judge: Honorable William Alsup

TABLE OF CONTENTS

Page

ORACLE'S OPPOSITION TO GOOGLE'S RULE 50(A) MOTION (ECF No. 1937) & RESPONSE TO COURT'S ORDER (ECF NO. 1934) 1

I. Google Has Not Conducted The Required Filtration Analysis 1

II. Google Cites No Cases To Support its Position Because Google's Argument Directly Contradicts Binding Precedent 5

III. Google's Motion Puts The Burden Of Proof On Oracle 5

IV. Google's Motion Is Wrong On the Facts 6

V. Google's Theory Was Not Properly Disclosed 7

ORACLE'S RESPONSE TO ECF NO. 1955 7

ORACLE'S OPPOSITION TO GOOGLE'S RULE 50(A) MOTION (ECF NO. 1937) & RESPONSE TO COURT'S ORDER (ECF NO. 1934)

Google's Rule 50(a) Motion should be denied. The motion is premised on a fundamental misunderstanding of the law and fails to cite the four appellate decisions that explain the very limited circumstances under which copying in order to learn about technical requirements for compatibility may constitute fair use. In order to grant Google's motion, the Court would have to shift to Oracle the burden of proof on Google's affirmative defense of fair use, specifically on whether Google's copying was technically necessary under fair-use factors two and three, and hold that an absence of proof in Google's case justifies granting judgment in *Google's* favor. Google's failure to meet even a basic burden of proof on this issue is readily apparent from the record. Google introduced no evidence that it copied in order to create compatibility with the Java programming language, much less the Java platform, during its case-in-chief, despite that fact that Google bears the burden of proof in Phase One. The only evidence relevant to Google's argument was submitted by Oracle, and it shows only that Google copied far more than necessary to use the Java programming language. On top of all of this, Google's motion is based on a theory that was not properly disclosed, and a theory that Google did not argue to the jury in either opening or closing. For all of these reasons, Google's motion should be denied.

I. GOOGLE HAS NOT CONDUCTED THE REQUIRED FILTRATION ANALYSIS

When the Federal Circuit remanded this case, it recognized that some components of the Java SE API packages "may be necessary for anyone to copy if they are to write programs in the Java language." *Oracle Am.*, 750 F.3d at 1377. The Federal Circuit also recognized, however, that the record on fair use, for which Google bears the burden of proof, did not provide a sufficient basis to determine whether certain aspects of the Java SE API packages were necessary to use the Java programming language. The Federal Circuit thus invited Google to conduct a "filtration analysis" on remand. *Id.* (citing *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1524–25 (9th Cir. 1992)).

Sega, the source of the Federal Circuit's "filtration analysis," established a narrow "reverse engineering" exception that recognizes a possible fair use defense for intermediate copying to understand *uncopyrightable* aspects of a computer program where the "unprotected aspects … cannot be examined without copying." *Sega*, 977 F.2d at 1526. The copying in *Sega* was intermediate because Accolade's final products were based only on "functional descriptions of the interface requirements [for the Sega Genesis computer]," but the final products neither included nor were based on "any of Sega's code." *Id.* at 1515.

The Ninth Circuit later revisited *Sega* in *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000). As in *Sega*, the copyrighted computer code in *Sony* contained some functional aspects that were not protected. *Id.* at 603. Information about Sony's code was not public, so the only way to determine which elements of the software were purely functional, and which were expressive, was to reverse engineer the object code (readable only by machines) into source code (readable by humans).[1] *Id.* The reverse engineering process required Connectix to make temporary computer-based copies of Sony's object code, which was "necessary to gain access to the unprotected functional elements within the program." *Id.* In *Sony*, as in *Sega*, the copying was only intermediate—Connectix and Accolade used what they learned about *functional* requirements to build a product that "does not itself contain infringing material." *Id.* at 606.

Key to the holdings in both *Sega* and *Sony* was that intermediate copies were made for the purpose of producing a product that was actually *compatible* with a preexisting system—in *Sega*, games compatible with the Sega Genesis computer were made, *Sega*, 977 F.2d at 1520, and in *Sony*, a computer program compatible with Sony's PlayStation games was made, *Sony*, 203 F.3d at 607 ("Connectix reverse-engineered the Sony BIOS to produce a product that would be *compatible* with games designed for the Sony PlayStation. We have recognized this purpose as a legitimate one under the first factor of the fair use analysis.") (emphasis added). Further, the copies were only intermediate, and none of the copied code made its way into either Accolade's or Connectix's final products. Under these narrow circumstances, the Ninth Circuit held that the intermediate copying, necessary to examine the functional aspects of an otherwise copyright-

[1] Analogous to the "bytecode" processed by the Java Virtual Machine and the Java "source code" in our case, respectively.

protected computer program, was fair use.

The Federal Circuit addressed a similar topic in *Atari Games Corp. v. Nintendo of Am. Inc.*, 975 F.2d 832 (Fed. Cir. 1992). There, the Federal Circuit recognized that "[w]hen the nature of a work requires intermediate copying to understand the ideas and processes in a copyrighted work, that nature supports a fair use for intermediate copying. Thus, reverse engineering object code to discern the unprotectable ideas in a computer program is a fair use." *Id.* at 843. Like the Ninth Circuit, however, the Federal Circuit recognized that this exception is limited, and "[a]ny reproduction of protectable expression must be *strictly necessary* to ascertain the bounds of protected information within the work." *Id.* (emphasis added). "This fair use did not give … more than the right to understand the [copyrighted] program and to distinguish the protected from the unprotected elements of the program." *Id.* at 844. Thus Atari's copying was not a fair use.

With this background in mind, it is clear that Google's copying satisfies none of the elements necessary to invoke the fair use exception for reverse engineering a computer program in order to identify functional components necessary for compatibility as that limited exception was carefully laid out in *Sega*, *Nintendo*, and *Sony*.

First, there is no evidence that Google copied the Java SE API packages in order "to gain access to the unprotected functional elements within the [Java SE API packages]." *Sony*, 203 F.3d at 603. Google has never claimed at any point during this litigation that it copied the Java API packages as research in order to determine what functional elements of the copied packages were necessary for compatibility with the Java language, and the evidence shows that Google could have "rewrit[ten] APIs using something that was a completely different package and class organization, that from a technical perspective that's true." Tr. 1268:7-11 (Astrachan); *see* Tr. 1849:15-18 (Agreed Fact) ("Sun/Oracle could have written and organized the declaring code for the 37 Java API packages in any number of ways and still have achieved the same functions."). Google concedes as much in its motion, recognizing that the evidence shows Google copied not to filter out copyrighted expression so that it only copied functional aspects of the Java API packages, but instead copied "the ones that would be useful," and ones that "'made sense' for the Android platform, based on the needs of a smartphone platform, " Mot. at 3. This is because "[t]he choice to use the 37 APIs was not a requirement of the Java programming language." Tr. 1268:12-14 (Astrachan) ("A. That's correct."); Tr. 1546:14-16 (D. Schmidt) ("[T]here was nothing in the Java Language, there was nothing in the Java Language specification, there was nothing in the programmer convention that forced the design of these libraries to have the same structure, sequence and organization.").

Second, Google did not need to copy anything to understand the technical requirements of the Java language specification for the Android API libraries. Google could have read a book without copying a single line of protected expression. Tr. 1464:20-25 (Reinhold) ("And other than being in the book in front of you, the Java Language Specification, Third Edition is there any other way that parts of the Java API could be required by the Java programming language? A. No. If it's not mentioned in the book, then it's not – it's not required by the language."). Google's copying is therefore nothing like the copying in *Sega*, *Sony*, and *Nintendo* where it was *necessary* to copy protected expression to access unprotected functional components of the computer programs.

Third, it is undisputed that Google's Android product is not compatible with Java SE, Tr. 1231:8-25 (Astrachan), meaning that Google's use was not for the "legitimate" purpose "to produce a product that would be compatible" with an existing software program. *Sony*, 203 F.3d at 607; Tr. 1231:25 (Astrachan) (Android and Java SE "platforms aren't compatible.").

Fourth, this is not a case where Google tried to comply with the reverse engineering cases and missed the mark. Google copied 11,500 lines of code where it needed to copy only 170 in order to use the Java programming language—or over 67 times what it needed to copy based on the Java language specification.

The law and the record are clear: The Ninth Circuit's reverse engineering cases have no applicability to this case, and Google needs to prove up its fair use case under the four statutory factors. Google knows this, which is why its motion does not cite or rely on *Sega*, *Sony*, *Nintendo*, or *Oracle Am.*, none of which support Google's motion.

II. GOOGLE CITES NO CASES TO SUPPORT ITS POSITION BECAUSE GOOGLE'S ARGUMENT DIRECTLY CONTRADICTS BINDING PRECEDENT

Google's argument is unmoored from any applicable case law, and Google does not cite a single case in its motion that supports what amounts to the elimination of Oracle's copyrights in the declaring code and SSO of the copied Java SE API packages.[2] Ninth Circuit case law is clear. Computer programs are copyrightable, and the fact that they consist of functional and expressive elements simultaneously does not deprive them of protection, or make copying them qualify as fair use. *Wall Data Inc. v. L.A. Cnty. Sheriff's Dep't*, 447 F.3d 769, 780 (9th Cir. 2006) (rejecting fair use defense) ("Although the RUMBA software products are not purely creative works, copyright law nonetheless protects computer software"). Google has not cited a single contrary case, or a case holding that difficulty in filtering functional aspects of a copyrighted work from the expressive aspects justifies any copying beyond the limited, intermediate copying approved in *Sega*, *Sony*, and *Nintendo*. Indeed, the facts in the reverse engineering cases show that making the intermediate copies necessary to conduct the filtration analysis was quite difficult, requiring steps such as "peeling" the layers off of silicon ROM chips in order to copy object code (the 1's and 0's readable by computers), *Nintendo*, 975 F.2d at 836, or studying source code in order to separate the functional elements of a computer program from the expressive elements, *Sega*, 977 F.2d at 1514-15. But the reverse engineering exception still applies to only intermediate copying, and no case holds that a defendant can simply throw up its hands and infringe an entire work if her or she finds the reverse engineering process too difficult (or in Google's case, fails to even try in the first place).

III. GOOGLE'S MOTION PUTS THE BURDEN OF PROOF ON ORACLE

It is undisputed that Google has the burden of proof on fair use. *See Monge v. Maya Magazines, Inc.*, 688 F.3d 1164, 1170 (9th Cir. 2012); ECF No. 1950 (Final Jury Charge) at 17. The Federal Circuit expressly invited the parties to conduct a "filtration analysis" to determine which code was technically necessary to copy. Oracle provided that analysis to the jury; Google

[2] Google cites only *Sony Corp. of America v. Universal Studios, Inc.*, 464 U.S. 417, 448 (1984), for the vague proposition that "fair use is 'an equitable rule of reason,'" in support of its ongoing effort to expand the doctrine beyond all established bounds. Google Mot. at 3.

did not. Google oddly claims that Dr. Reinhold's analysis, which benefits Google by eliminating 170 lines of copied code from the case, is unreliable. Even if it were, eliminating Dr. Reinhold's analysis from the case means Google has *no evidence* that *any* portion of the Java SE API packages was necessary in order use the Java programming language. Google would be even further from meeting its burden if Dr. Reinhold's analysis were rejected on the second and third fair use factors than it is already is. *See* Mot. at 1. It was Google's burden to prove that its copying was fair use because it was required to use the Java programming language.

IV. GOOGLE'S MOTION IS WRONG ON THE FACTS

It is not true that the declaring code that was technically necessary to copy due to the constraints of the language is indistinguishable from the rest of the declaring code. *See* Google Mot. at 2-3. Dr. Reinhold testified that "the Java programming language [is] defined by a specification," Tr. 1462:6-8, introduced at trial as TX 984, Tr. 1462:9-11. Dr. Reinhold explained that "there are lines or partial lines of declaring code" which "the Java Language Specification mandates" be included in a Java language API implementation. Tr. 1464:9-13. But beyond those limited number of lines mentioned in the Java language specification, "[i]f it's not mentioned in the book [TX 984], then it's not – it's not required by the language." Tr. 1464:24-25. All in all, Dr. Reinhold concluded that 170 lines of code were technically necessary to copy. Tr. 1464:19.[3] If Google wanted to copy only what was technically necessary, it could have simply looked in the language specification book, and written its own API libraries on top of the 170 lines of API declaring code required by the Java language specification.

But there is no evidence that Google even attempted to read and follow the Java language specification book by taking steps to limit its copying to elements required by the language specification. In fact, Google's own motion states that Google selected what to copy not by trying to adhere to the minimum necessary for the language, but by choosing "the ones that 'made sense' for the Android platform, based on the needs of a smartphone platform." Mot. at 1. The most that anyone has ever suggested *might* possibly be necessary to copy from Java SE in order to

[3] Google's attempts to question the reliability of Dr. Reinhold's analysis should be rejected. Google knew that Dr. Reinhold revised his analysis, but stipulated to the revised analysis anyhow, and did not cross examine Dr. Reinhold on the issue when he testified.

use the Java programming language are the three so-called "core" packages identified by the Federal Circuit. *Oracle Am.* 750 F.3d at 1365. Even by that measure, Google copied 34 Java API packages too many.

V. GOOGLE'S THEORY WAS NOT PROPERLY DISCLOSED

But the Court does not even need to wade into this analysis. Google did not mention the concept that the technically necessary declaring code was indistinguishable from the rest of the declaring code in response to Oracle's interrogatory requesting the bases for any fair use contention. Google's First Suppl. Resp. & Obj. to Oracle's Interrogs., Set 6, No. 40 at 9-15. Nor was this novel legal theory included in the Joint Proposed Pretrial Order, ECF 1709, or the Court's Final Pretrial Conference Order, ECF No. 1760. Accordingly, Google cannot sandbag Oracle with this undisclosed theory now, *after* the close of evidence. No one—not Google, not the Federal Circuit, not any other court—has ever so much as hinted at Google's new theory of fair use. The only reason Google filed its Rule 50(a) motion is because the Court ordered Google to file on this topic. When Google ran out of space in its opposition to Oracle's Rule 50(a) motion, it had to file a separate motion. Thus, Google cannot cite a single *case* that supports its motion for judgment *as a matter of law*, other than to quote the vague general proposition that fair use is an "equitable rule of reason."

The Court should deny Google's motion for judgment as a matter of law.

* * *

ORACLE'S RESPONSE TO ECF NO. 1955

The Court requested that the parties "advise what evidence in the trial record addresses the causal mechanism by which Android (which was given away for free) itself added to Google's advertising revenue." ECF No. 1955. In particular, the Court asked "what, if anything, was special about Android itself (as opposed to other platforms) leading to the use of the Google search engine (and thus to greater advertising revenue)." *Id.* In short, Android massively increased Google's advertising revenue and preserved Google's advertising revenue streams.

No Relevance To Factors One Or Four. This issue is not relevant to either factor one or factor four. As discussed above and in Oracle's motion, Android is massively commercial and is "hugely profitable." TX 951; Tr. 421:25 (E. Schmidt). Whether Google might have also been able to make money in some other way if Android did not exist is not relevant to *Android's* commerciality, which is the inquiry under factor one. Tr. 2205:4-5 (Jury Charge) ("in evaluating the first statutory factor, the extent of the commercial nature of the *accused use* must be considered") (emphasis added) Similarly, whether Google could have earned revenue in some other way that did not harm Sun (i.e., by *using* Sun's mobile platform rather than displacing it) is irrelevant to factor four. Factor four considers the harm to markets and potential markets from Google's actual infringement or from widespread conduct of the sort engaged in by Google. Tr. 2211:10-2213:11 (Jury Charge). Replacement of Android by some non-infringing platform is irrelevant to this inquiry.

The Evidence Shows Android Increases Google's Revenue. Google's own documents show that "Android has a Direct Revenue Impact" on Google. TX 1061 at 15. The reason is that "people who use Android search twice as much as everything else." Tr. 421:4-7 (E. Schmidt). And "[n]ot only is [sic] there more searches, and there's more ads, but it's also more lucrative." Tr. 421:21-23 (E. Schmidt). That means "there's more revenue associated with Android searches." Tr. 421:8-10 (E. Schmidt). Google's documents and witnesses stated it bluntly: "Search + Android = Huge." TX 5183 at 7 (Rubin); Android is "hugely profitable." TX 951; Tr. 421:25 (E. Schmidt).

Evidence submitted after Google rested its case only reinforces the conclusion that Android increases Google's profits. Android has made over $42 billion in revenue. Tr. 1762:8-9 (Jaffe). And an internal Google presentation describes Android as supporting a "$43B / year ecosystem" that also "protect[s] us [Google] from losing control." TX 190 at 8. Against all this evidence, there is nothing in the record suggesting that Android does *not* increase Google's revenue. That is why Google went through the trouble of quickly getting Android to the market in the first place, and why Google has never argued that Android does not increase its profits.

How Android Increases Profits. There are several reasons why Android (as opposed to some other platform) increases Google's profits. As stated above, "people who use Android search twice as much as everything else." Tr. 421:4-7 (E. Schmidt). Additionally, Schmidt explained that Google does not need to share as much of its advertising revenue when Google controls the platform. He testified that if customers are "using Android operating systems the revenue that we share and the searches are shared with the operator *but not with anybody else.*" Tr. 421:11-14 (emphasis added). By contrast, as discussed below, when Google advertises on other company's platforms, Google does have to share the revenue. To take one example, one proposed Google/Sun deal term was: "rev share on platform-enabled mobile ads." TX 339 at 6; *see also* Tr. 786:8-787:18 (Rubin discussing document). Because of Android, Google earned all that money instead of having to share some of it with Sun.

Android also ensured that Google would not be "locked out" of the mobile market. TX 31 at 12 ("Why did Google invest in Android? … Don't get locked out!"). Rubin "thought at the time Microsoft was [Google's] main competitor." Tr. 643:14-15. He also viewed Symbian as a competitor. Tr. 755:17-18. If some other platform had filled the void instead of Android, there was no guarantee that platform would have used Google's search technology or, even if it did, that Google would have had as prominent a position on such platform. Tr. 1754:16-1756:2; 1761:3-14. What was special about Android was that it allowed Google to control the platform, and thus to ensure the dominance of its search engine on that platform. *See* Tr. 1701:14-19; 1836:21- 1837:5. The more Google's search engine is used, the more ad revenues are generated.

To the extent the Court's questions relate to damages, the details of how Android increases Google's profits will be fleshed out in more detail in Phase Two. Dr. Kearl's report provides a helpful overview of the evidence the Court can expect the jury to hear. Dr. Kearl describes how Android affects Google's "Traffic Acquisition Costs" or "TAC." TAC is the portion of ad revenue that Google pays to its partners in exchange for setting Google as the default search engine. Kearl Report ¶ 22.

Dr. Kearl explained: "The development and market success of Android was important to Google because (among other possible reasons) the success of Android prevented other mobile device platform operators (such as Apple and Microsoft) from directing web traffic on their devices away from Google and thereby being 'locked out' of the increasingly large mobile search and advertising business or using the threat to do so in order to negotiate a higher TAC (i.e., a higher share of the Google ad revenue for ads displayed on the other platform's devices)." Kearl Report ¶ 23. He went on: "One measure of the value of Android to Google is that it allows Google to pay lower TAC costs (and therefore keep a larger share of its ad revenue) for ads displayed on an Android device than what Google pays Apple or Microsoft or other non-Android platform device developers for ads displayed on an iPhone or Windows Mobile phone or other non-Android device." *Id.*

This is consistent with the evidence already in the record that Google did not want to get "locked out", TX 31, and that "if [customers are] using Android operating systems the revenue that we share and the searches are shared with the operator *but not with anybody else*," Tr. 421:11-14 (emphasis added). And Google's own documents will show [redacted] *See e.g.*, TX 4104 at 6-7 (not yet admitted).

That being said, whether Google could have made money without Android is not relevant to disgorgement because it is an analysis based on non-infringing alternatives. ECF No. 1784 at 7 ("Nor may Google offer Leonard's non-infringing alternatives analyses to rebut Oracle's causal nexus theory.)

Dated: May 25, 2016

Respectfully submitted,

Orrick, Herrington & Sutcliffe LLP

By: */s/ Peter A. Bicks*
Peter A. Bicks

Counsel for ORACLE AMERICA, INC.